????

Harley Eades III

Computer Science
Augusta University
harley.eades@gmail.com

Abstract. TODO

1 Introduction

2 A Ternary Semantics for SAND Attack Trees

References

1. Ravi Jhawar, Barbara Kordy, Sjouke Mauw, Saša Radomirović, and Rolando Trujillo-Rasua. Attack trees with sequential conjunction. In Hannes Federrath and Dieter Gollmann, editors, *ICT Systems Security and Privacy Protection*, volume 455 of *IFIP Advances in Information and Communication Technology*, pages 339–353. Springer International Publishing, 2015.

Appendix

.1 SSG Semantics

In this appendix I show that the category of source-sink graphs defined by Jhawar et al. [1] is symmetric monoidal. First, recall the definition of source-sink graphs and their homomorphisms.

Definition 1. *A* ***source-sink graph*** *over* B *is a tuple* $G = (V, E, s, z)$*, where* V *is the set of vertices,* E *is a multiset of labeled edges with support* $E^* \subseteq V \times \mathsf{B} \times V$*,* $s \in V$ *is the unique start,* $z \in V$ *is the unique sink, and* $s \neq z$*.*

Suppose $G = (V, E, s, z)$ *and* $G' = (V', E', s', z')$*. Then a* ***morphism between source-sink graphs****,* $f : G \to G'$*, is a graph homomorphism such that* $f(s) = s'$ *and* $f(z) = z'$*.*

Suppose $G = (V, E, s, z)$ and $G' = (V', E', s', z')$ are two source-sink graphs. Then given the above definition it is possible to define sequential and non-communicating parallel composition of source-sink graphs where I denote disjoint union of sets by $+$ (p 7. [1]):

Sequential Composition :
$G \rhd G' = ((V \setminus \{z\}) + V', E^{[s'/z]} + E', s, z')$

Parallel Composition :
$G \odot G' = ((V \setminus \{s, z\}) + V', E^{[s'/s, z'/z]} + E', s', z')$

It is easy to see that we can define a category of source-sink graphs and their homomorphisms. Furthermore, it is a symmetric monoidal category were parallel composition is the symmetric tensor product. It is well-known that any category with co-products is symmetric monoidal where the co-product is the tensor product.

I show here that parallel composition defines a co-product. This requires the definition of the following morphisms:

$$
\begin{array}{l}
\mathsf{inj}_1 : G_1 \to G_1 \odot G_2 \\
\mathsf{inj}_2 : G_2 \to G_1 \odot G_2 \\
\langle f, g \rangle : G_1 \odot G_2 \to G
\end{array}
$$

In the above $f : G_1 \to G$ and $g : G_2 \to G$ are two source-sink graph homomorphisms. Furthermore, the following diagram must commute:

$$
\begin{tikzcd}
 & G \arrow[dl, "f"'] \arrow[dr, "g"] & \\
G_1 \arrow[r, "\mathsf{inj}_1"'] & G_1 \odot G_2 \arrow[u, "\langle f,g \rangle"] & G_2 \arrow[l, "\mathsf{inj}_2"]
\end{tikzcd}
$$

Suppose $G_1 = (V_1, E_1, s_1, z_1)$, $G_2 = (V_2, E_2, s_2, z_2)$, and $G = (V, E, s, z)$ are source-sink graphs, and $f : G_1 \to G$ and $g : G_2 \to G$ are source-sink graph morphisms – note that $f(s_1) = g(s_2) = s$ and $f(z_1) = g(z_2) = z$ by definition. Then we define the required co-product morphisms as follows:

$$
\begin{array}{l}
\mathsf{inj}_1 : V_1 \to (V_1 \setminus \{s_1, z_1\}) + V_2 \\
\mathsf{inj}_1(s_1) = s_2 \\
\mathsf{inj}_1(z_1) = z_2 \\
\mathsf{inj}_1(v) = v, \text{ otherwise} \\
\\
\mathsf{inj}_2 : V_2 \to (V_1 \setminus \{s_1, z_1\}) + V_2 \\
\mathsf{inj}_2(v) = v \\
\\
\langle f, g \rangle : (V_1 \setminus \{s_1, z_1\}) + V_2 \to V \\
\langle f, g \rangle(v) = f(v), \ \text{where } v \in V_1 \\
\langle f, g \rangle(v) = g(v), \ \text{where } v \in V_2
\end{array}
$$

It is easy to see that these define graph homomorphisms. All that is left to show is that the diagram from above commutes:

$$\begin{aligned}(\mathsf{inj}_1;\langle f,g\rangle)(s_1) &= \langle f,g\rangle(\mathsf{inj}_1(s_1))\\ &= g(s_2)\\ &= s\\ &= f(s_1)\end{aligned}$$

$$\begin{aligned}(\mathsf{inj}_1;\langle f,g\rangle)(z_1) &= \langle f,g\rangle(\mathsf{inj}_1(z_1))\\ &= g(z_2)\\ &= z\\ &= f(z_1)\end{aligned}$$

Now for any $v \in V_1$ we have the following:

$$\begin{aligned}(\mathsf{inj}_1;\langle f,g\rangle)(v) &= \langle f,g\rangle(\mathsf{inj}_1(v))\\ &= f(v)\end{aligned}$$

The equation for inj_2 is trivial, because inj_2 is the identity.